一种改进的警示传播算法求解 Max-SAT 问题 *

吴宇翔[a]，王晓峰[a, b]，丁红胜[a†]，于　卓[a]

(北方民族大学 a. 计算机科学与工程学院; b. 图像图形智能处理国家民委重点实验室, 银川 750021)

摘　要：Max-SAT 问题是 SAT 问题的优化版本，目标是在给定的子句集中，找到一组变元赋值，使得满足子句数最多，该问题是典型的 NP-hard 问题。随着大数据和人工智能的深度发展，过去原有的算法已不再适用，设计新的求解算法或对已有的求解算法进行优化是目前研究的热点。本文针对警示传播算法求解随机 Max-3-SAT 问题的局限性，提出了一种基于变元权值计算的警示传播算法，结合随机游走算法，给出一种新型算法 WWP+WalkSAT，通过改进求解的局限性，更好地得到一组有效的初始解，从而提高算法的局部搜索能力。利用 2016 年 Max-SAT 国际竞赛部分基准实例，将 WWP+WalkSAT 算法与 8 种局部搜索算法进行精度方面的对比实验。实验结果表明 WWP+WalkSAT 算法有较好的性能。

关键词：可满足性问题；最大可满足性问题；警示传播算法；局部搜索算法

中图分类号：TP301　　doi: 10.19734/j.issn.1001-3695.2022.01.0023

Improved warning propagation algorithm for solving max-sat problem

Wu Yuxiang[a], Wang Xiaofeng[a, b], Ding Hongsheng[a†], Yu Zhuo[a]

(*a. School of Computer Science & Engineering*, *b. The Key Laboratory of Images & Graphics Intelligent Processing of State Ethnic Affairs Commission*, *North Minzu University*, *Yinchuan* 750021, *China*)

Abstract: The Max-SAT problem is an optimized version of the SAT problem. The goal is to find a set of variable assignments in a given set of clauses so that the maximum number of clauses is satisfied. This problem is a typical NP-hard problem. With the in-depth development of big data and artificial intelligence, the original algorithms in the past are no longer applicable. Designing a new solution algorithm or optimizing the existing solution algorithm is the current research hotspot. Aiming at the limitations of the information dissemination algorithm for solving the random Max-3-SAT problem, this paper proposes a warning dissemination algorithm based on the calculation of variable weights. Combined with the random walk algorithm, a new algorithm WWP+WalkSAT is formed, which is solved by improvement The limitation of, it is better to get a set of effective initial solutions, thereby improving the local search ability of the algorithm. Using some benchmark examples of the Max-SAT International Competition in 2016, the WWP+WalkSAT algorithm and 8 local search algorithms are used for accuracy comparison experiments. The experimental results show that the WWP + walksat algorithm has good performance.

Key words: satisfiability problem; max-sat problem; warning propagation algorithm; local search algorithm

0　引言

组合优化问题在运筹学、离散数学和计算机科学中都有着非常重要的作用，并在国防、交通、医疗、通信等领域有着广泛的应用，其中常见的组合优化问题包括背包问题(Knapsack)、旅行商问题(Travelling salesman problem, TSP)、车辆路径规划问题(Vehicle Routing Problem,VRP)、最大团问题(Maximum Clique Problem, MCP)、最小顶点覆盖问题(Minimum vertex cover, MVC)、最大可满足性问题(Maximum Satisfiability, Max-SAT)[1~6]等。而最大可满足性问题是一个典型的 NP 难问题，是可满足性问题(Satisfiability,　SAT)[7]的优化版本。给定一个命题合取范式(CNF)，　CNF 是由一组子句合取构成，每一个子句由一组变元析取构成，Max-SAT 问题是寻找一组赋值使得满足子句数目最多。Max-SAT 在组合拍卖、车辆调度、排课安排等现实问题中有着广泛的应用，同时图论中的最大团问题和顶点支配集也可以转换成 Max-SAT 问题进行求解。

为了解决 Max-SAT 问题，研究人员提出许多有效算法，主要分为完备性算法和非完备性算法两种。完备性算法是可以找到问题的精确解，但在求解大规模问题时，由于其复杂度是指数级，很难在合理时间找到结果。近年来，人们主要在分支策略、推理规则、下界估计[8~10]三方面进行改进，进而出现了许多有效的算法，具有代表性的有：WmaxSatz[11]、MiniMaxSat[12]等。相对于完备算法而言，非完备性算法则是能短时间内找到大规模问题的最优解，从而提高了求解效率，但缺点则是不能保证解的准确性。主流的非完备性算法是近似算法、信息传播算法、局部搜索算法[13~15]。

信息传播算法是一种启发式信息传递算法，起源于统计物理学，该类算法通过边缘概率计算的方法，已应用在许多领域。特别是对于命题公式的可满足性问题，有三种基于因子图的信息传播算法，即警示传播算法(warning propagation, WP)、信念传播算法(belief propagation, BP)和调查传播算法(survey propagation, SP)。目前，在求解随机 SAT 问题上信息传播算法是最为有效的方法，并且在图着色、最大流、LDPC 编码等许多组合优化问题中取得了不错的成果。

然而，信息传播算法的收敛性和有效性一直是研究的重

收稿日期：2022-01-14；**修回日期：**2022-03-15　　**基金项目：**国家自然科学基金资助项目(62062001)；宁夏自然科学基金资助项目(2020AAC03214)；北方民族大学研究生创新项目(YCX21086)

作者简介：吴宇翔(1997-)，男，宁夏银川人，硕士研究生，主要研究方向为算法设计与分析；王晓峰(1980-)，男，甘肃会宁人，副教授，博士，主要研究方向为机器学习、算法设计与分析；丁红胜(1977-)男(通信作者)，甘肃甘谷人，副教授，主要研究方向为数字图像处理，算法分析与设计(78950228@qq.com)；于卓(1997-)，男，宁夏银川人，硕士研究生，主要研究方向为算法设计与分析.

点。收敛性是指：信息传播算法两次的信息迭代值不发生变化；有效性是指：信息传播算法能够有效的求解问题。文献[16~18]分析了信息传播算法的收敛性和有效性,并给出了算法收敛的充分条件。文献[19]则是基于具体的实例产生模型分析了信息传播算法的收敛性,并给出了算法收敛的概率条件。

进一步研究发现，在随机 3-SAT 问题中，子句个数 m 和变元个数 n 的比值称约束密度 α，即 $\alpha=m/n$，它不仅对公式的可满足性产生影响，还对可满足性公式的求解难易程度有着非常重要的影响。随着约束密度 α 的增大，当 $3.52<\alpha<4.48$ 时，会发生相变。在相变范围以外的可满足性实例均为易解实例，高概率是可满足的；在相变点附近的的实例属于难解实例，高概率是不可满足的。虽然信息传播算法对于求解难解实例十分有效，但在求解相变点以外的易解实例方面，信息传播算法不能收敛。

针对于此问题，文本设计了一种新型变元权值计算的警示传播算法 WWP+WalkSAT，通过变元权值计算对警示传播算法进行改进，可以得到一组有效的初始解，再结合随机游走算法进行局部搜索。对该算法的性能表现进行实验分析，在相变点附近实例和相变点以外实例均与其余局部搜索算法进行实验对比。实验结果表明：对于求解任何区域的实例，WWP+WalkSAT 在求解精度方面平均优于其他局部搜索算法。

1　基础知识

最大可满足性问题是指：给定一组命题变元 $x_i\in X$，由这些变元形成一组子句，构成 *CNF* 公式，使得 *CNF* 公式中满足子句的个数最多，即，使得不满足的子句个数最少。Max-SAT 问题的数学模型如式(1)和式(2)：

Maximize：

$$\sum_{a\in F} w_a\cdot z_a \tag{1}$$

Subject to：

$$\forall a\in F:\ \sum_{i\in S_a^+} x_i+\sum_{i\in S_a^-}(1-x_i)\ge z_a$$

$$z_a\in\{0,1\}\ \ x_i\in\{0,1\} \tag{2}$$

文字的集合(Literal) $X=\{x_1,x_2,x_3,\ldots,x_n\}$：每一个布尔变元 $x_i\in X$，变元 x_i 取正时表示正文字；变元 x_i 取反时表示负文字。

子句的集合(Clause) $C=\{C_1,C_2,C_3,\ldots,C_m\}$：对 m 个不同的子句形成一个子句集合，每一个子句由一个或多个文字组成，子句与子句之间通过析取运算连接，且仅当子句中至少有一个文字为 1 时，该子句满足，反之子句不满足。每一个子句中文字的个数叫做这个子句的长度。当子句中只有一个文字出现时，称该子句为单子句。

合取范式(CNF 公式)：由若干个子句合取构成，即 $C_1\wedge C_2\wedge C_3\wedge\ldots\wedge C_m$，当且仅当 CNF 公式的每一个子句都满足时，称合取范式 CNF 可满足。

因子图(Factor Graph)：因子图也称二部图，图中包含两类节点：一类是变元节点(图中用圆环表示，记住(1，2，3，4;⋯)),另一类是函数节点(图中中方框表示,记住(a,b,c;⋯))。给一个 CNF 公式 $F=\{C_1,C_2,C_3,\ldots,C_m\}$,公式中由 $x_1,x_2,x_3,\ldots,x_n$ 个变元构成，其中所有变元对应的是变元节点，变元析取构成的子句对应的是函数节点。图中出现的虚边和实边代表着 *CNF* 公式的两种不同的含义：实边：子句中代表正文字 x_j，虚边：子句中代表负文字 $\neg x_j$。

例 1：$F=\{(x_1\vee\neg x_2\vee\neg x_3)\wedge(\neg x_1\vee x_2\vee\neg x_4)\wedge(\neg x_1\vee x_3\vee x_5)\wedge(\neg x_3\vee x_4\vee x_5)\}=\{a\wedge b\wedge c\wedge d\}$ 其对应的因子图如图 1 所示。

$V(a)=V^+(a)+V^-(a)$ 表示子句 a 中出现的变元集合 $V(a)=V^+(a)+V^-(a)$。其中：$V^+(a)$ 表示子句 a 中变元为正的集合；$V^-(a)$ 表示子句 a 中变元为负的集合；$V(a)\backslash i$ 表示子句 a 中除变元 i 以外其余变元的集合：在图 1 中，$V(a)=\{x_1,x_2,x_3\}$ $V^+(a)=\{x_1\}$，$V^-(a)=\{x_2,x_3\}$。

$V(j)$ 表示含有变元 x_j 的子句集合 $V(j)=V^+(j)+V^-(j)$，其中 $V^+(j)$ 表示变元 x_j 为正的子句集合；$V^-(j)$ 表示变元 x_j 为负的子句集合；$V(j)\backslash a$ 表示除了子句 a 以为，其余含有变元 x_j 的子句集合。在图 1 中，$V(1)=\{a,b,c\}$，$V^+(1)=\{a\}$，$V^-(1)=\{b,c\}$。

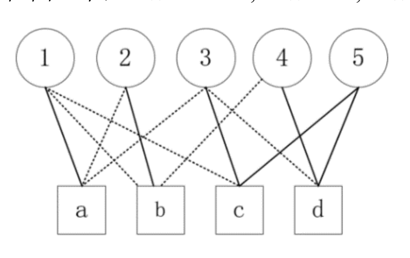


图 1　因子图

Fig. 1　Factor graph

J_j^a：对于子句 a 中的变元 x_j 的标识符，变元在子句中的取值，由式(3)表示为

$$J_j^a=\begin{cases}-1, & x_j\in a\\ 1, & \neg x_j\in a\end{cases} \tag{3}$$

定义两个一致性集合 $V_a^u(j)$ 和 $V_a^s(j)$

$V_a^u(j)$ 表示除子句 a 以外，含有变元 x_j 且变元 x_j 的取值方向与其子句 a 中取值不一致的子句集；$V_a^s(j)$ 表示除子句 a 以外，含有变元 x_j 且变元 x_j 的取值方向与其子句 a 中取值一致的子句集；集合 $V_a^u(j)$ 和 $V_a^s(j)$ 如图 2 所示。

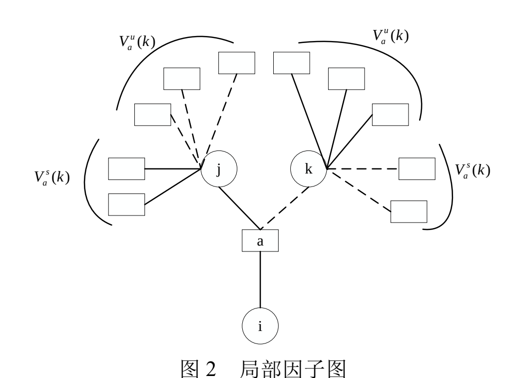


图 2　局部因子图

Fig. 2　Part of factor graph

2　基于变元权值计算的警示传播算法

基于变元权值计算的警示传播算法 WWP+WalkSAT 是在警示传播算法[20]上进行改进，文献[16]分析警示传播算法的收敛性，通过引入参数，将算法中的信息取值从 $\{0,1\}$ 松弛为 $[0,1]$，利用向量空间中压缩函数的性质，给出警示传播算法收敛的一个充分条件，并为警示传播算法性能提供了理论依据。WWP+WalkSAT 是在此理论基础上，引入了变元权值计算[21]，从中得到一组有效的初始解，其目的是选出权值最高的文字，使得满足子句个数最多，从而减少无效的子句传播，加快搜索过程。原本的警示传播算法在求解难解实例十分有效，但对于易解实例会出现不收敛现象。对于此问题，该算法在迭代次数结束时，计算局部警示信息变化最小时的警示值，再进行变元权值计算。最后对其初始解采用随机游走的局部搜索算法，可得到更优解，提从而高其算法性能。WWP+WalkSAT 打破了原先警示传播算法求解 Max-SAT 问题的局限性，可用于求解任何区域的实例，同时在求解精度上也有所提高。

2.1　警示传播算法(Warning Propagation)

通过信息传递而设计的信息传播算法，对于求解可满足性问题具有良好的有效性，警示传播算法是一个迭代运算的算法，在每一次迭代过程中，对于因子图的每条边 (a,i)，都

会获得一个警示信息，表示在因子图中一个函数节点 a 传递给变元 i 一个布尔信息，记作 $u_{a\to i}\in\{0,1\}$。$u_{a\to i}$ 的迭代方程如式(4)所示。

$$u_{a\to i}(t+1)=\prod_{j\in V(a)\setminus i}\theta(-J_j^a(\sum_{b\in V(j)\setminus a}J_j^b u_{b\to j}(t))) \tag{4}$$

其中 t 是迭代次数，$\theta(x)$ 是一个截尾函数，定义为：$\theta(x)=\begin{cases}1, & x>0\\ -1, & x\le 0\end{cases}$，由此公式可以说明，当 $u_{a\to i}=1$ 表示对于子句 a 的满足性由变元 x_i 来决定，当 $u_{a\to i}=0$ 表示子句 a 的满足性不能完全由变元 x_i 决定，即子句 a 的满足性由其余变元的取值来决定。

对于每一个变元 x_i，需要计算一个局部腔域 H_i 和一个冲突域 $c_i\in\{0,1\}$，其中局部腔域和冲突域式(5)和式(6)所示。

$$H_i=-\sum_{b\in V(i)}J_i^b u_{b\to i}^*=\sum_{b\in V^+(i)}u_{b\to i}^*-\sum_{b\in V^-(i)}u_{b\to i}^* \tag{5}$$

$$c_i=\begin{cases}1, & if\,(\sum_{b\in V^+(i)}u_{b\to i}^*)(\sum_{b\in V^-(i)}u_{b\to i}^*)>0.\\ 0, & otherwise.\end{cases} \tag{6}$$

当冲突域 $c_i=1$ 时，即 $(\sum_{b\in V^+(i)}u_{b\to i}^*)(\sum_{b\in V^-(i)}u_{b\to i}^*)>0$，则表示对于变元 x_i，受到了两个子句的约束，变元 x_i 在两个子句中的取值不一致，因此变元 x_i 的取值发生了冲突。如：子句 a 告诉变元 x_i 取 1，子句 b 告诉变元 x_i 取 0。当冲突域 $c_i=0$ 时，即 $(\sum_{b\in V^+(i)}u_{b\to i}^*)(\sum_{b\in V^-(i)}u_{b\to i}^*)=0$，则表示对于变元 x_i 的取值没有冲突发生，变元 x_i 的具体取值可以通过局部腔域来决定，如式(7)所示。

$$x_i=\begin{cases}1, & \sum_{b\in V^+(i)}u_{b\to i}^*>0,\ \sum_{b\in V^-(i)}u_{b\to i}^*=0\\ \{0,1\}, & \sum_{b\in V^+(i)}u_{b\to i}^*=\sum_{b\in V^-(i)}u_{b\to i}^*=0\\ 0, & \sum_{b\in V^+(i)}u_{b\to i}^*=0,\ \sum_{b\in V^-(i)}u_{b\to i}^*>0\end{cases} \tag{7}$$

如果 $H_i>0$，变元 $x_i=1$；如果 $H_i<0$，变元 $x_i=0$；否则变元 x_i 暂时不赋值。

算法 1 警示传播算法(WP)

输入：CNF 因子图 G；最大迭代次数 $t_{\max}$；精度 ε。

输出：未收敛或收敛，输出警示信息值 $u^*_{a\to i}$

Begin

当 $t=0$ 时，对于因子图的每个边 (a,i)，以 1/2 的概率随机初始化警示信息 $u_{a\to i}(0)\to\{0,1\}$ //随机初始化

$For\ t=1\ to\ t_{max}$

通过字典的顺序，对图中每一条边 $(a,i)\in E$ 进行遍历，采用警示更新算法 UPDATE-WP 对 $u_{a\to i}$ 进行更新，即 $u_{a\to i}(t-1)\to u_{a\to i}(t)$

当图中所有的边 $(a,i)\in E$，若 $\sum_{(a,i)\in E}|u_{a\to i}(t)-u_{a\to i}(t-1)|<\varepsilon$，说明已收敛，$u_{a\to i}(t)=u_{a\to i}(t-1)$，停止循环，返回 $u^*_{a\to i}(t)$

IF $t=t_{\max}$

寻找 $t^*=\arg\min_t\sum_{(a,i)\in E}|u_{a\to i}(t)-u_{a\to i}(t-1)|$

返回：未收敛，并将 $u^*_{a\to i}(t)=u_{a\to i}(t^*)$， //把 t^* 时刻的信息值保留并赋值给警示值

算法 2 警示信息值更新算法(UPDATE-WP)

输入：警示信息值集合 $\{u_{b\to j}:b\in V(j)\setminus a\}$，对所有的 $j\in V(a)\setminus i$

输出：新的 $u_{a\to i}$

Begin

IF: $V(a)\setminus i=\phi$，则返回 $u_{a\to i}=1$

ELSE:

for $j\in V(a)\setminus i$

IF $V(j)\setminus a=\phi$，则 $h_{j\to a}=0$

Else $h_{j\to a}=(\Sigma_{b\in V_+(j)\setminus a}u_{b\to j})-(\Sigma_{b\in V_-(j)\setminus a}u_{b\to j})$

返回 $u_{a\to i}=\Pi_{j\in V(a)\setminus i}\theta(J_j^a h_{j\to a})$

算法 3 警示信息赋值指派算法

输入：CNF 因子图 G

输出：输出一组变元赋值指派 $\{x_1,x_2,x_3\ldots x_n\}$

Begin

运行 WP 算法

计算所有的局部腔域 H_i 和冲突域 c_i

IF $c_i=0$，则对变元进行权值计算

IF $c_i=1$

IF $H_i>0$，则 $x_i=1$

IF $H_i<0$，则 $x_i=0$

ELSE $H_i=0$，则对变元进行权值计算

返回变元赋值指派

2.2 变元权值计算

在迭代运算时，基础的警示传播算法只对冲突域 $c_i=0$，且局部腔域 $H_i>0$ 的变元赋值为 1，局部腔域 $H_i<0$ 的变元赋值为 0，而对于冲突域 $c_i=1$ 的变元和 $c_i=0$ 且 $H_i=0$ 的变元无法赋值。在求解最大可满足性问题中，变元赋值的好坏直接影响着整个算法的求解效率，正确的变元赋值可以高效求解。本文提出了一种新的启发式变元权值计算，通过权值计算来确定该变元的赋值。

对于未赋值的每一个变元都有一个正或负的权值，这取决于该变元的正文字和负文字出现的差异，变元的权值代表着变元为正或为负的程度。如果权值为正，则代表变元在整个 CNF 公式中更多地以正文字的形式出现，而不是负文字的形式出现，反之亦然。权重计算如式(8)：

$$W_{var}=\frac{Number(PosLit)-Number(NegLit)}{Number(Clause_{var})} \tag{8}$$

其中 $Number(PosLit)$ 表示变元正文字出现次数，$Number(NegLit)$ 表示变元负文字出现次数，$Number(Clause_{var})$ 表示变元在子句中出现的次数。

算法 4 变元权值算法

输入：CNF 公式

输出：变元 x_i 的赋值

Begin

For 对每一个未赋值的变元 i：

初始变元权值 $W_i=0$，找到包含变元 i 的所有子句 $Clause_i$ 并记录总子句数 $Number_i$

For 对每一个子句 $Clause_i$ 中的变元

If $i>0$，$PosLit++$

Else $i<0$，$NegLit++$

$$W_i=\frac{PosLit-NegLit}{Number_i}$$

IF $W_i>0$，则返回 $x_i=1$

Else 返回 $x_i=0$

2.3 局部搜索算法

在局部搜索算法中，搜索过程占用整个算法的大部分时间，普遍随机搜索算法的初始解都是随机选取，会造成初始解的随机性，好的初始解可以高效的找到结果，不好的初始解会浪费很长时间且降低了算法的性能，同时还会重复地访问一些解，出现循环的现象。本文提出的变元权值的警示传播算法，可以有效得到一组初始解，对后续进行局部搜索有很大帮助。

基于变元权值的警示传播算法得到一组变元的赋值，由此构造了一组初始解。初始解得到的子句数不一定是最优的，需要局部搜索算法进行深一步搜索。对于 Max-SAT 问题，目的是找到满足子句数最多的一组赋值，将得到的一组变元赋

值放入到 WalkSAT 中进行搜索。

3　实验结果与分析

为了准确评估本文提出的新型变元权值计算的警示传播算法能在解决 Max-SAT 问题时起到很好的效果，本文对算法 WWP+WalkSAT、WalkSAT、SA、GA、VNS-GA、CCLS2akms、CCEHC、Optirise6-in 和 HS-Greedy 进行了测试。其中 WalkSAT、SA、GA、VNS-GA 为经典的启发式算法，CCLS2-akms、CCEHC、Optirise6-in 和 HS-Greedy 为 2016 年 Max-SAT Evalution 国际竞赛的求解器。实验结果使用了 2016 年 Max-SAT Evalution 国际竞赛中随机类的数据集。

表 1 给出了 SA、WalkSAT、WWP+WalkSAT、GA、VNS-GA 几种算法之间的统计比较结果。采用 s3v70c1000、s3v80c1000 两种数据集(3-SAT 实例，变元个数 70 和 80，子句个数 1000),对满足最大子句个数的结果进行了对比实验。表中可以看出，在 70 个子句中的 10 个实例中，有 7 个实例都是取得最好的结果，有 3 个实例略差一些。在 80 个子句中，只有 4 个实例取得最好结果，其余 6 个效果稍差一些。但是可以明显看出 WWP+WalkSAT 的效果远远高于 WalkSAT，且满足的子句个数也大大提高。

表 1　SA、WalkSAT、WWP+WalkSAT、GA 和 VNS-GA 在易解实例中实例对比

Tab.1　Comparison of SA, WalkSAT, WWP+WalkSAT,GA and VNS-GA in easily solved instances

Instance	The number of satisfy clauses				
	SA	WalkSAT	WWP+WalkSAT	GA	VNS-GA
s3v70c1000-1	**950**	929	**950**	921	946
s3v70c1000-2	948	924	**955**	931	952
s3v70c1000-3	945	927	**950**	922	949
s3v70c1000-4	948	930	**948**	928	**948**
s3v70c1000-5	951	926	951	933	**955**
s3v70c1000-6	946	926	**948**	929	945
s3v70c1000-7	**955**	929	953	930	943
s3v70c1000-8	947	931	**953**	931	950
s3v70c1000-9	955	925	956	930	**958**
s3v70c1000-10	949	930	**954**	928	952
s3v80c1000-1	953	926	**955**	953	953
s3v80c1000-2	951	925	942	**953**	**953**
s3v80c1000-3	**958**	930	949	946	955
s3v80c1000-4	942	929	**953**	951	950
s3v80c1000-5	954	927	953	**955**	**955**
s3v80c1000-6	950	934	**958**	956	956
s3v80c1000-7	948	927	953	**957**	**957**
s3v80c1000-8	952	929	950	955	**955**
s3v80c1000-9	950	932	953	958	**959**
s3v80c1000-10	947	931	**959**	952	957

表 2~3 是将 SA、WalkSAT、WWP+WalkSAT、CCEHC、CCLS2akms、Optirise6-in 和 HS-Greedy 进行了测试。实验结果使用了 2016 年 Max-SAT Evalution 竞赛中随机类的数据集。其中表 2 中采用 3-SAT 实例，变元数 70、90、110 个，子句数 700、800、900、1000、1100 的数据集。每一种实例有 10 组数据。实验结果采用平均精度计算表示，即最大可满足子句数/实例子句数，再对 10 组数据求平均值。实验表明：对于 WWP+WalkSAT 来说，精度达到了 95%~99%，而 CCEHC 的精度在 95%到 97%。且在 15 种数据集中，有 12 个数据集 WWP+WalkSAT 在求解精度方面排名第一，并且与 CCEHC、HS-Greedy 结果相差不大，但远高于了 SA、WalkSAT、CCLS2akms、Optirise6-in 几种算法。说明 WWP+WalkSAT 算法对于求解易解算例十分有效，而且优于普通的局部搜素算法。表 3 中，实验采用 HG-3SAT-V300-C1000、HG-3SAT-V250-C1200 两种数据集(3SAT 实例，300 个变元，1000 和 1200 个子句，$\alpha=4$）。实验结果表明，算法 CCLS2akms 是无法求解该类问题的，WWP+WalkSAT 的求解精度均达到了 98~99%左右，而 CCEHC 的精度稳定在 99%左右。WWP+WalkSAT 还是略差异 CCEHC，但是性能远远好于 SA、WalkSAT、Optirise6-in 和 HS-Greedy 几种算法。在难解算例中，WWP+WalkSAT 稍差于 CCEHC 算法，但在易解实例中，WWP+WalkSAT 求解性能好于 CCEHC 算法。由此可以说明 WWP+WalkSAT 算法对求解随机 Max-3-SAT 的问题十分有效。

除此之外，通过 5 个数据集对算法 WWP+WalkSAT 和 WalkSAT 在迭代次数方面进行了对比实验。图 3~7 表明，两者均进行 1000 次迭代，结果表明除了数据集 S3v110c1000 中初始结果 WWP+WalkSAT 略差于 WalkSAT，其余四个数据集一开始 WWP+WalkSAT 就优于 WalkSAT。这是因为 WalkSAT 的初始解是随机赋值，当初始解赋值好的时候会优于 WWP+WalkSAT，而 WWP+WalkSAT 的初始解是通过变元权值计算得出的，所以一开始的满足子句个数就会优于 WalkSAT。而随着迭代次数的增加，WWP+WalkSAT 满足子句的个数远远优于 WalkSAT。尽管在数据集 S3v90c900 中出现 WalkSAT 高于 WWP+WalkSAT 的现象，但是最后 WWP+WalkSAT 还是取得了不错的结果。虽然在精度上取得了不错的效果，但是 WWP+WalkSAT 算法的运行时间会差于 WalkSAT 算法，这也是该算法今后需要改进的地方。

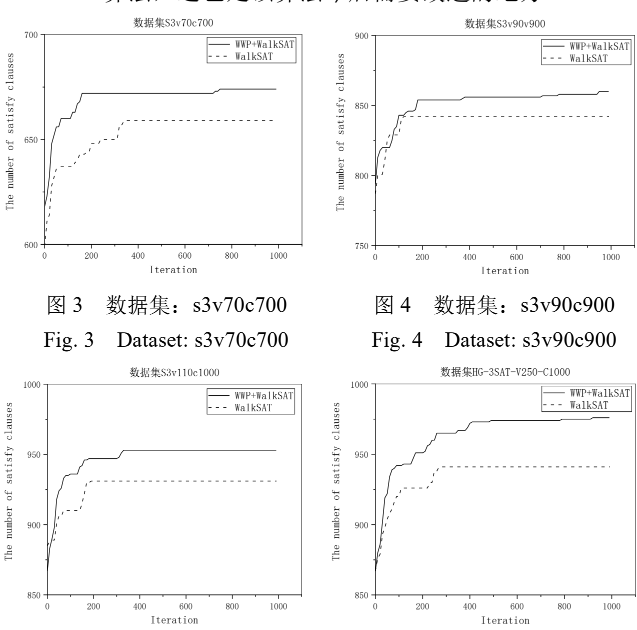


图 3　数据集：s3v70c700　　图 4　数据集：s3v90c900

Fig. 3　Dataset: s3v70c700　　Fig. 4　Dataset: s3v90c900

图 5　数据集：s3v110c1000　　图 6　数据集：HG-3SAT-V250-C1000

Fig. 5　Dataset: s3v110c1000　　Fig. 6　Dataset: HG-3SAT-V250-C1000

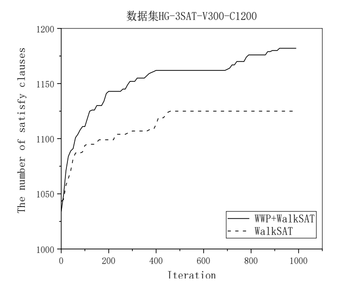


图 7　数据集：HG-3SAT-V300-C1200

Fig. 7　Dataset: HG-3SAT-V300-C1200

4　结束语

本文设计基于变元权值计算的警示传播算法 WWP+

WalkSAT，对其算法进行了精度分析，实验表明该算法对易解 Max-3-SAT 问题还是难解 Max-3-SAT 问题都具有较好的优势，而且也打破了信息传播算法求解的局限性，其结果对于以后的理论研究提供了很大的参考价值。与此同时该算法也可用于解决不同领域的实际问题，在组合拍卖、车辆调度、机器人路径规划、资源配置和网络社交分析等方面有广阔的应用前景。但是由于使用了基于变元权值计算的警示传播算法增加了整个计算的复杂度因而消耗了时间，后续的研究将会在算法的变元剪枝策略以及其他减少时间效率的方法入手研究。

表 2　SA、WalkSAT、WWP+WalkSAT、CCEHC、CCLS2akms、Optirise6-in 和 HS-Greedy 在易解实例中实验对比

Tab. 2　Comparison of SA、walksat、WWP+walksat、CCEHC、CCLS2akms、Optirise6-in and HS-Greedyin easily solved instances

Instance	Number	Precision(The number of satisfy clauses / clauses)						
		SA	WalkSAT	WWP+WalkSAT	CCLS2akms	CCEHC	Optirise6-in	HS-Greedy
s3v70c700	10	95%	92%	96%	93%	97%	86%	96%
s3v70c800	10	95%	93%	96%	93%	96%	87%	96%
s3v70c900	10	95%	92%	96%	93%	95%	86%	95%
s3v70c1000	10	95%	93%	95%	93%	96%	88%	94%
s3v70c1100	10	94%	93%	95%	93%	94%	88%	94%
s3v90c700	10	96%	94%	97%	59%	97%	86%	96%
s3v90c800	10	96%	94%	96%	59%	97%	86%	96%
s3v90c900	10	95%	93%	96%	59%	96%	87%	95%
s3v90c1000	10	95%	93%	96%	59%	96%	88%	96%
s3v90c1100	10	95%	93%	96%	59%	96%	90%	96%
s3v110c700	10	97%	95%	97%	60%	97%	86%	97%
s3v110c800	10	97%	95%	99%	60%	97%	86%	97%
s3v110c900	10	97%	95%	98%	60%	97%	88%	97%
s3v110c1000	10	97%	95%	97%	60%	96%	88%	96%
s3v110c1000	10	97%	95%	98%	60%	97%	87%	97%

表 3　SA、WalkSAT、WWP+WalkSAT、CCEHC、CCLS2akms、Optirise6-in 和 HS-Greedy 在难解实例中实验对比

Tab. 3　Comparison of SA、walksat、WWP+walksat、CCEHC、CCLS2akms、Optirise6-in and HS-Greedyin difficultly solved instances

Instance	The number of unsatisfy clauses						
	SA	WalkSAT	WWP+WalkSAT	CCEHC	CCLS2akms	Optirise6-in	HS-Greedy
HG-3SAT-V250-C1000-1	23	64	9	5	N/A	138	12
HG-3SAT-V250-C1000-2	25	66	10	6	N/A	122	13
HG-3SAT-V250-C1000-3	27	64	9	6	N/A	129	10
HG-3SAT-V250-C1000-4	28	61	8	5	N/A	115	12
HG-3SAT-V250-C1000-5	24	69	10	6	N/A	110	11
HG-3SAT-V250-C1000-6	27	60	4	6	N/A	128	14
HG-3SAT-V250-C1000-7	23	66	6	6	N/A	120	14
HG-3SAT-V250-C1000-8	26	64	10	5	N/A	123	13
HG-3SAT-V250-C1000-9	22	65	9	7	N/A	115	12
HG-3SAT-V250-C1000-10	27	68	5	6	N/A	114	13
HG-3SAT-V300-C1200-1	29	68	13	6	N/A	156	18
HG-3SAT-V300-C1200-2	28	67	10	6	N/A	155	20
HG-3SAT-V300-C1200-3	25	70	13	7	N/A	151	15
HG-3SAT-V300-C1200-4	31	72	4	6	N/A	134	16
HG-3SAT-V300-C1200-5	25	65	4	6	N/A	126	15
HG-3SAT-V300-C1200-6	27	71	3	7	N/A	151	14
HG-3SAT-V300-C1200-7	25	68	6	6	N/A	141	16
HG-3SAT-V300-C1200-8	28	60	5	7	N/A	133	15
HG-3SAT-V300-C1200-9	31	72	5	7	N/A	149	17
HG-3SAT-V300-C1200-10	26	69	7	6	N/A	155	18

参考文献：

[1] 王熙照，贺毅朝. 求解背包问题的演化算法 [J]. 软件学报, 2017, 28 (1): 1-16. (Wang Xizhao, He Yichao. Evolutionary algorithms for knapsack problems [J]. Journal of Software, 2017, 28 (1): 1-16.)

[2] Gunduz M, Aslan M. DJAYA: A discrete Jaya algorithm for solving traveling salesman problem [J]. Applied Soft Computing, 2021, 105: 107275.

[3] Mousaei A, Taghaddos H, Nekouvaght Tak A, *et al.* Optimized Mobile Crane Path Planning in Discretized Polar Space [J]. Journal of Construction Engineering and Management, 2021, 147 (5): 04021036.

[4] 王晓峰，于卓，赵健，等. 基于大规模图的最大团问题算法分析 [J/OL]. 计算机工程: 1-15 [2022-03-11]. DOI: 10. 19678/j. issn. 1000-3428. 0063092.

[5] Masato, Suzuki, Yoshiyuki, *et al.* Statistical mechanics of the minimum vertex cover problem in stochastic block models. [J]. Physical review. E,

2019, 100 (6-1): 62101-62101.

[6] 何琨，郑迥之. 最大可满足性问题的算法研究综述 [J]. 华中科技大学学报：自然科学版，2022，50 (02): 82-95. DOI: 10. 13245/j. hust. 220214.

[7] Aiello A, Burattini E, Massarotti A. The Complexity of Theorem Proving Procedures [J]. Rairo Informat Théor, 1977, 11 (1): 75-82.

[8] 刘燕丽，黄飞，张婷. 基于环型扩展推理规则的 MaxSAT 完备算法 [J]. 南京大学学报：自然科学，2015，51 (04): 762-771. DOI: 10. 13232/j. cnki. jnju. 2015. 04. 014.

[9] Li Chumin, Xu Zhenxing, Coll J, *et al.* Combining Clause Learning and Branch-and-Bound for MaxSAT [C]// Proc of the International Conference on Principles and Practice of Constraint Programming. Montpellier: Springer, 2021: 38: 1-38: 18

[10] Morgado A, F Heras, Liffiton M, *et al.* Iterative and core-guided MaxSAT solving: A survey and assessment [J]. Constraints, 2013, 18 (4): 478-534.

[11] Li Chumin, Felip Manyà, Nouredine Ould Mohamedou N O, *et al.* Resolution-based lower bounds in MaxSAT [J]. Constraints, 2010, 15 (4) .

[12] Heras F, Larrosa J, Oliveras A. MiniMaxSat: a new weighted Max-SAT solver [C]// Springer Berlin Heidelberg. Springer Berlin Heidelberg, 2007.

[13] Johnson D S. Approximation algorithms for combinatorial problems [J]. Journal of Computer & System Sciences, 1974.

[14] Park S, Shin J. Convergence and correctness of max-product belief propagation for linear programming [J]. Siam Journal on Discrete Mathematics, 2017, 31 (3): 2228-2246.

[15] Chu Yi, Luo Chuan, Cai Shaowei, *et al.* Empirical investigation of stochastic local search for maximum satisfiability [J]. 中国计算机科学前沿：英文版, 2019, 13 (1): 13.

[16] 王晓峰，许道云. 警示传播算法收敛的充分条件 [J]. 软件学报，2016, 27 (12): 3003-3013. DOI: 10. 13328/j. cnki. jos. 004940.

[17] 王晓峰，许道云，杨德仁，姜久雷，李强，刘欣欣. 可满足性问题中信念传播算法的收敛性分析 [J]. 软件学报, 2021, 32 (05): 1360-1372. DOI: 10. 13328/j. cnki. jos. 005844.

[18] Shi X, Schonfeld D, Tuninetti D. Message error analysis of loopy belief propagation [C]// 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2010: 2078-2081.

[19] 王晓峰，许道云，韦立. 随机可满足实例集上警示传播算法的收敛性 [J]. 软件学报, 2013, 24 (01): 1-11.

[20] Braunstein A, Mezard M, Zecchina R. Survey propagation: an algorithm for satisfiability [J]. Random Structures & Algorithms, 2010, 27 (2): 201-226.

[21] Layeb A. A new greedy randomized adaptive search procedure for solving the maximum satisfiability [J]. International Journal of Operational Research, 2013, 17 (3): 1-17.